P1. Using the monoalphabetic cipher in Figure 8.3, encode the message "This is an easy problem." Decode the message "rmij'u uamu xyj."

Problem 1

The encoding of "This is an easy problem" is "uasi si my cmiw lokngch".
The decoding of "rmij'u uamu xyj" is "wasn't that fun".

P2. Show that Trudy's known-plaintext attack, in which she knows the (ciphertext, plaintext) translation pairs for seven letters, reduces the number of possible substitutions to be checked in the example in Section 8.2.1 by approximately 10^9.

Problem 2

If Trudy knew that the words "bob" and "alice" appeared in the text, then she would know the ciphertext for b,o,a,l,i,c,e (since "bob" is the only palindrome in the message, and "alice" is the only 5-letter word. If Trudy knows the ciphertext for 7 of the letters, then she only needs to try 19!, rather than 26!, plaintext-ciphertext pairs. The difference between 19! and 26! is 26*25*24...*20, which is 3315312000, or approximately 10^9.

P3. Consider the polyalphabetic system shown in Figure 8.4. Will a chosen-plaintext attack that is able to get the plaintext encoding of the message "The quick brown fox jumps over the lazy dog." be sufficient to decode all messages? Why or why not?

Problem 3

Every letter in the alphabet appears in the phrase "The quick fox jumps over the lazy brown dog." Given this phrase in a chosen plaintext attack (where the attacker has both the plain text, and the ciphertext), the Caesar cipher would be broken - the intruder would know the ciphertext character for every plaintext character. However, the Vigenere cipher does not alway translate a given plaintext character to the same ciphertext character each time, and hence a Vigenere cipher would not be immediately broken by this chosen plaintext attack.

P4. Consider the block cipher in Figure 8.5. Suppose that each block cipher T_i simply reverses the order of the eight input bits (so that, for example, 11110000 becomes 00001111). Further suppose that the 64-bit scrambler does not modify any bits (so that the output value of the mth bit is equal to the input value of the mth bit). (a) With $n = 3$ and the original 64-bit input equal to 10100000 repeated eight times, what is the value of the output? (b) Repeat part (a) but now change the last bit of the original 64-bit input from a 0 to a 1. (c) Repeat parts (a) and (b) but now suppose that the 64-bit scrambler inverses the order of the 64 bits.

Problem 4

a) The output is equal to 00000101 repeated eight times.
b) The output is equal to 00000101 repeated seven times + 10000101.
c) We have (ARBRCR)R = CBA, where A, B, C are strings, and R means inverse operation. Thus:
   1. For (a), the output is 10100000 repeated eight times;
   2. For (b), the output is 10100001 + 10100000 repeated seven times.

P5. Consider the block cipher in Figure 8.5. For a given "key" Alice and Bob would need to keep eight tables, each 8 bits by 8 bits. For Alice (or Bob) to store all eight tables, how many bits of storage are necessary? How does this number compare with the number of bits required for a full-table 64-bit block cipher?

Problem 5

a) There are 8 tables. Each table has 28 entries. Each entry has 8 bits.
   number of tables * size of each table * size of each entry = 8*28* 8= 214 bits

b) There are 264 entries. Each entry has 64 bits. 271 bits

P6. Consider the 3-bit block cipher in Table 8.1. Suppose the plaintext is 100100100. (a) Initially assume that CBC is not used. What is the resulting ciphertext? (b) Suppose Trudy sniffs the ciphertext. Assuming she knows that a 3-bit block cipher without CBC is being employed (but doesn't know the specific cipher), what can she surmise? (c) Now suppose that CBC is used with IV = 111. What is the resulting ciphertext?

Problem 6

a) 100100100 ==> 011011011
b) Trudy will know the three block plaintexts are the same.
c) c(i) = KS(m(i) XOR c(i-1))
c(1) = KS(100 XOR 111) = KS (011) = 100
c(2) = KS(100 XOR 100) = KS (000) = 110
c(1) = KS(100 XOR 110) = KS (010) = 101

P8. Consider RSA with $p = 5$ and $q = 11$.

a. What are n and z?

b. Let e be 3. Why is this an acceptable choice for e?

c. Find d such that $de = 1 \pmod{z}$ and $d < 160$.

d. Encrypt the message $m = 8$ using the key (n, e). Let c denote the corresponding ciphertext. Show all work. *Hint:* To simplify the calculations, use the fact:

Problem 8

p = 5, q = 11
a) n = p*q = 55, z = (p-1)(q-1) = 40
b) e = 3 is less than n and has no common factors with z.
c) d = 27
d) m = 8, me = 512, Ciphertext c= me mod n = 17

P9. In this problem, we explore the Diffie-Hellman (DH) public-key encryption algorithm, which allows two entities to agree on a shared key. The DH algorithm makes use of a large prime number p and another large number g less than p. Both p and g are made public (so that an attacker would know them). In DH, Alice and Bob each independently choose secret keys, S_A and S_B, respectively. Alice then computes her public key, T_A, by raising g to S_A and then taking mod p. Bob similarly computes his own public key T_B by raising g to S_B and then taking mod p. Alice and Bob then exchange their public keys over the Internet. Alice then calculates the shared secret key S by raising T_B to S_A and then taking mod p. Similarly, Bob calculates the shared key S' by raising T_A to S_B and then taking mod p.

a. Prove that, in general, Alice and Bob obtain the same symmetric key, that is, prove $S = S'$.

b. With $p = 11$ and $g = 2$, suppose Alice and Bob choose private keys $S_A = 5$ and $S_B = 12$, respectively. Calculate Alice's and Bob's public keys, T_A and T_B. Show all work.

c. Following up on part (b), now calculate S as the shared symmetric key. Show all work.

d. Provide a timing diagram that shows how Diffie-Hellman can be attacked by a man-in-the-middle. The timing diagram should have three vertical lines, one for Alice, one for Bob, and one for the attacker Trudy.

Problem 9

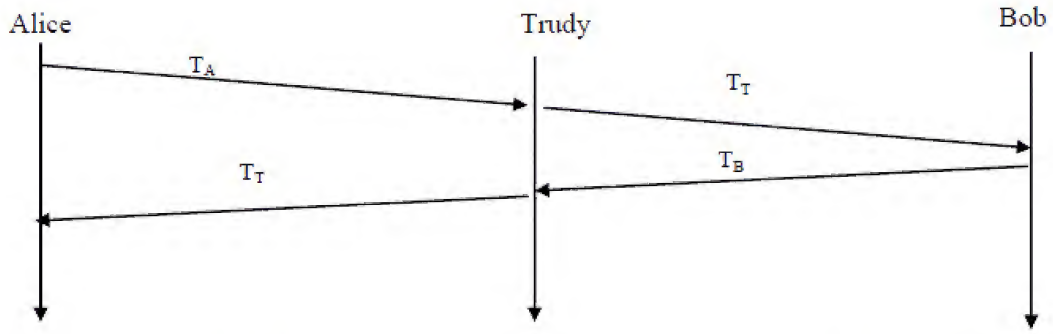
	Alice	Bob
secrect key:	S_A	S_B
public key:	$T_A = (g \text{^} S_A) \bmod p$	$T_B = (g \text{^} S_B) \bmod p$
shared key:	$S = (T_B \text{^} S_A) \bmod p$	$S' = (T_A \text{^} S_B) \bmod p$

a) $S = (TB \text{^} SA) \bmod p = ((g \text{^} SB \bmod p) \text{^} SA) \bmod p = (g \text{^} (SBSA)) \bmod p$
$= ((g \text{^} SA \bmod p) \text{^} SB) \bmod p = (TA \text{^} SB) \bmod p = S'$

(b and c) $p = 11$, $g = 2$

	Alice	Bob
secrect key:	$S_A = 5$	$S_B = 12$
public key:	$T_A = (g \text{^} S_A) \bmod p = 10$	$T_B = (g \text{^} S_B) \bmod p = 4$
shared key:	$S = (T_B \text{^} S_A) \bmod p = 1$	$S' = (T_A \text{^} S_B) \bmod p = 1$

d)



The Diffie-Hellman public key encryption algorithm is possible to be attacked by man-in-the-middle.

1. In this attack, Trudy receives Alice's public value (T_A) and sends her own public value (T_T) to Bob.
2. When Bob transmits his public value (T_B), Trudy sends her public key to Alice (T_T).
3. Trudy and Alice thus agree on one shared key (S_{AT}) and Trudy and Bob agree on another shared key (S_{BT}).

4. After this exchange, Trudy simply decrypts any messages sent out by Alice or Bob by the public keys S_{AT} and S_{BT}.